'93 한국자동제어학술회의논문집 (1993.10.20~22)

은닉층에 대한 의미부여를 통한 학습에 대한 연구

°기세훈, 안상철, 권욱현
서울대학교

A Study for Learning Neural-Network using Internal Representation

°Kee SaeHun, Ahn SangChul, Kwon WookHyun
Seoul National University

Abstract

Beacuse of compiecity, neurai network is difficuit tc iearn. So if internai representation[1] can be performed successfuiiy, it is possibie to use prceptron iearing ruie. As a resuit, iearning is easier. Therefore the method of internai representationis appiied to the "XOR" probiem, and the "spirais" probiem. And then using the above resuits, the structure of neurai network for computing is embodied.

1. 서론

신경망은 지금까지 여러 분야에서 많은 연구가 이루어 지고 있다. 그리고 그것들의 많은 부분이 패턴인식이나, 음성인식등 기존의 알고리즘등으로는 수행하기 힘든 일들이다 [2][3]. 그로 인해 신경망은 계속적인 연구가 진행되게 되었다. 그런데 이 신경망은 다른 알고리즘과 달리 학습이라는 과정을 필요로 하게 된다. 이 학습의 진행과정은 경우에 따라 문제를 일으키고, 학습이 곤란한 경우도 발생하였다. 이러한 원인은 상당 부분이 신경망이 다층으로 이루어져 있으므로 생겨난다.

위에서 언급하였듯이 신경망은 대부분 여러 계층으로 이루어져 있다. 이 다수의 계층에서 입력을 받아들이는 입력계층과 출력을 나타내는 출력층만이 그 각각의 의미를 지니게 된다. 이러한 이유로 인공지능에서 사용하는 것[4][5]과 같은 각 계층에 대한 의미 부여를 통해, 여러 계층으로 이루어진 신경망의 학습에 있어서 이를 두개의 계층으로 이루어져 있는 신경망의 학습으로 바꾸어 줌으로써 출력의 오차 값이 최대가 아닌 극소값에 이르는 경우를 배제할 수 있게 된다.

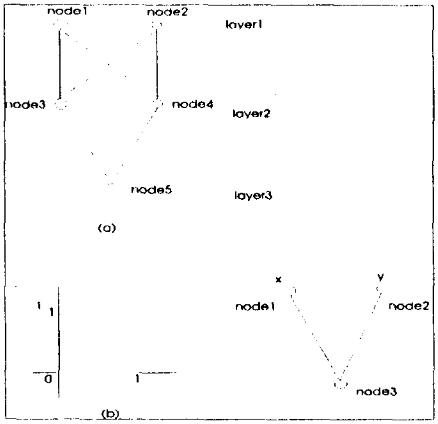
또한 각 계층의 노드에 의미가 무엇인지를 알수 있으므로 적당한 갯수의 계층의 수와 노드의 갯수를 정할 수 있게 된다. 또 이러한 사실의 적절한 이용을 통하여 패턴 인식이 아닌 수학적인 계산이 가능한 신경망이나, 시스템의 제어에 적당한 신경망에 대한 필요 조건과 구조를 결정할 수 있게 된다. 물론 이러한 절차를 거치지 않고 직접 여러가지 학습 방법을 이용하여 다층의 신경망을 학습 시키는 것이 효과적인 경우도 많을 것이다. 그러나 어느 정도 그 구조가 갖추어진 상태에서 좀더 정확한 결과를 원하는 경우나, 계층의 숫자나 노드의 갯수를 정하는 것이 어려운 경우[6]에는 여기서 제안하고자 하는 은닉층에 대한 의미부여가 도움이 될 수 있을 것이다.

지금까지의 은닉층에 대한 의미부여는 대부분 은닉층이 가질 수 있다고 판단되어지는 것들의 집합을 형성하고, 이 중 실제로 필요한 집합의 원소들을 골라내면서 신경망의 학습을 수행하는 것이였다[1]. 이에 대해 여기서는 패턴인식의 경우, 수식연산의 경우에 따른 은닉층에 대한 의미부여 방법을 제안한다. 패턴인식의 경우에는 신경망의 선형분리의 성질을 주로 이용하고, 수식연산의 경우에는 수의 양, 음의 분리를 이용한다.

2. 은닉층에 대한 의미부여 방법

은닉층에 대한 의미부여는 인공지능에서 서치를 위해 구현한 트리구조를 응용한 것이다. 인공지능에서 작업계획의 작성이나, 어떤 상황에서의 개념들간의 관계설정, 또는 게임의 과정을 나타내는 것들을 트리 구조로서 표현한다[7]. 이 구조는 신경망의 구조와 상당히 흡사한 구조를 가진다. 반면에 이 두가지는 노드간의 연결에 있어서 차이를 가지게 된다. 즉 인공 지능에서의 노드간의 관계는 관계의 유무를 나타낼 뿐 관계의 정도를 나타내지는 못하게 된다. 반면 신경망에서는 노드간의 관계를 차등적으로 나타낼 수 있다[12]. 이 두가지 구조에서의 차이는 퍼지개념을 사용함으로써 좀더 잘 살펴 볼 수 있다. [8]

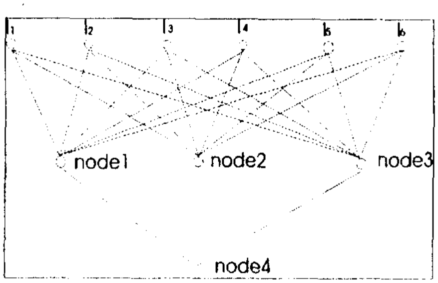
그러면 여기서 가장 간단한 예인 "XOR" 문제에 대해서 은닉층에 대한 의미부여를 하여 본다. 그 구조는 그림 1.1과 같다.



<그림 1.1 "XOR"에 대한 의미부여에 대한 구조>

은닉층의 첫번째 노드는 $x=1$ and $y=0$임을 나타내고, 두번째 노드는 $x=0$ and $y=1$임을 나타내는 구조이다. 이는 그림 1.1(b)에서 보이는 바와 같이 선형 분리되어야 하는 부분을 분리해 내는 것이다. 그리고 출력층에서는 은닉층의 노드중 꼭 하나가 ON(즉 1)이되는 순간 ON이 된다. 여기서 사용될수 있는 함수는 단위 함수, 시그모이드 함수등 일반적으로 사용이 되는 전달 함수가 거의 대부분 사용이 가능하다. 그런데 전달 함수에 종형 함수를 사용한다면, 은닉층이 없어질 수도 있게 된다. 그 구조는 그림 1.1(c)와 같다. 그 이유는 종형 함수는 단위함수나 시그모이드 함수가 한번에 하나의 선형 분리만 할 수 있는 반면에 종형 함수는 평행한 두개의 선형 분리선을 형성할 수 있다. 이러한 종형 함수의 경우에는 평균값과 분산, 이 두가지를 학습 대상으로 하여야 한다. 그리고 복잡한 함수의 모양으로 인하여 학습과 계산에 더 많은 시간을 요구하게 된다. 이러한 단점을 보완 하기위해 삼각형, 또는 사각형모양의 함수(그림 1.1(d))를 사용 할 수도 있다.

위의 문제의 확장인 패리티 문제도 그림1.2과 같은 형태로써 신경망을 구현해 볼 수 있다. 여기서 은닉층의 각 노드는 1의 갯수가 1, 3, 5임을 나타낸다.

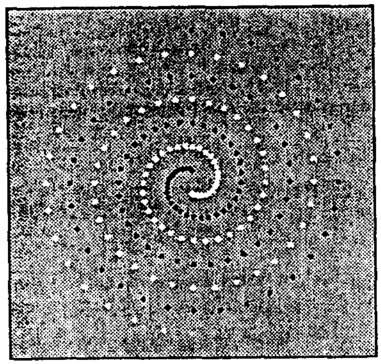


<그림 1.2 패리티 문제에 대한 은닉층 의미부여>

3. 패턴 인식에 사용되는 신경망에의 은닉층 의미 부여 방법

신경망의 중요한 특징중의 하나가 선형 분리에 있다. 이 특성을 고려하면 패턴인식을 하는 신경망에의 은닉층 의미 부여가 가능하다. 즉 선택이 되어야 할 부분을 선형 분리를 통하여 구별해 낼 수 있다는 성질을 사용하면 된다.[9][10][11] 여기서는 전달함수는 종형 함수를 사용하고자 한다.

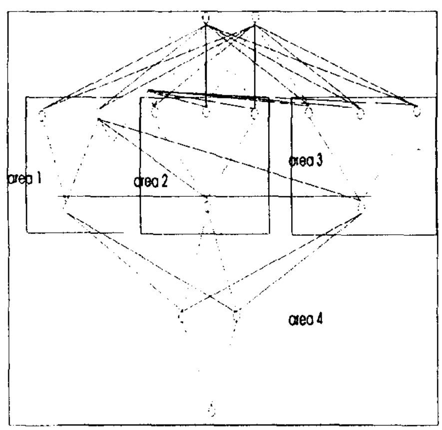
이제 부터 예를 통해 자세한 내용을 상술한다. 여기서 사용하는 예는 스파이어럴 문제이다[13]. 문제는 다음과 같다.



<그림 3.1 스파이어럴 문제의 학습 패턴>

이 문제를 해결하기 위해서 그림 3.2와 같은 신경망의 구조를 제안한다.

그림3.2에서 나타나 있듯이 5계층으로 구성된 신경망을 사용한다. 여기서 제안하고자 하는 은닉층에 대한 의미부여는



<그림 3.2 스파이어럴 문제에 사용되는 신경망의 구조>

평면 좌표계를 이용한다. 즉 그림 3.1의 가로축을 x, 세로축을 y, 그리고 중심의 좌표를 (0,0)이라고 둔다. 이러한 기준으로 부터 학습 패턴을 위한 프로그램 소스가 만들어 진다.[13]

이상의 문제 형성을 바탕으로 그림 3.2의 은닉층의 각각의 노드에 대한 의미를 만들어 아래에 정리 하였다. (아래에서의 정리 방식은 서로 관계있는 의미를 가지는 노드들 별로 구별한 것이다.)

1계층 1노드 : 입력의 x축에 해당하는 값
1계층 2노드 : 입력의 y축에 해당하는 값

2계층 1노드 : x축 값의 양, 음의 구분
2계층 2노드 : y축 값의 양, 음의 구분
3계층 1노드 : 2, 4분면을 구분해낸다. 즉 2계층의 1, 2 노드중 하나만 ON이 된 상태에서 ON이 된다.

2계층 3노드 : 1사분면에서 흰색 점들 중 첫번째 영역을 분리한다. 또 3사분면에서 검은색 점들중 첫번째 영역을 분리해 낸다.
2계층 4노드 : 1사분면에서 흰색 점들 중 두번째 영역을 분리한다. 또 3사분면에서 검은색 점들중 두번째 영역을 분리해 낸다.
2계층 5노드 : 1사분면에서 흰색 점들 중 세번째 영역을 분리한다. 또 3사분면에서 검은색 점들중 세번째 영역을 분리해 낸다.
3계층 2노드 : 1사분면의 흰색영역에 대해서는 ON상태를, 그리고 3사분면에서는 검은색영역에 대해서는 OFF상태를 출력으로 한다.

2계층 6노드 : 2사분면에서 흰색 점들 중 첫번째 영역을 분리한다. 또 4사분면에서 검은색 점들중 첫번째 영역을 분리해 낸다.
2계층 7노드 : 2사분면에서 흰색 점들 중 두번째 영역을 분리한다. 또 4사분면에서 검은색 점들중 두번째 영역을 분리해 낸다.
2계층 8노드 : 2사분면에서 흰색 점들 중 세번째 영역을 분리한다. 또 4사분면에서 검은색 점들중 세번째 영역을 분리해 낸다.
3계층 2노드 : 2사분면의 흰색영역에 대해서는 ON상태를, 그리고 4사분면에서는 검은색영역에 대해서는 OFF상태를 출력으로 한다.

4계층 1노드 : 1, 3분면에 대한 출력만을 내보낸다.
4계층 2노드 : 2, 4분면에 대한 출력만을 내보낸다.

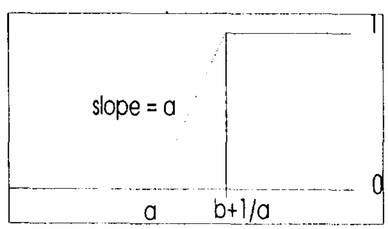
5계층 1노드 : 4계층의 어느 하나만이가 ON이 되었을 때만 ON.

이상에서 보였듯이 각 노드의 의미부여는 주어진 패턴을 선형분리의 원리에 따라 영역을 설정하고 설정된 구간 간의 공통점이 있다면 그들은 가능하면 같은 노드로 통일될 수 있도록 한다. 그리고 혼돈의 여지가 있는 경우에는 그들간의 분리를 위한 노드를 확실하게 삽입하여 주어야 한다. 위의 경우에서는 사분면을 구분하는 부분과 1,3 사분면에 대한 부분, 2,4분면에 관한 부분으로 분리하여 구성된 후, 4계층에서 이들 간에 관계를 설정하여 주고 있다. 또 앞에서 구성한 "XOR" 문제와 패리티 문제에 대한 신경망이 중간중간에 삽입되어 사용되어짐을 볼 수 있다. 즉 기존에 구성된 신경망이 사용됨으로써 학습의 효율을 높일 수 있을 것이다.

여기서 보인 패턴인식 문제의 경우에는 계층의 숫자나 노드의 갯수를 신경망 구성자가 쉽게 결정할 수 없는 문제이다. 그렇게 때문에 은닉층의 의미부여가 신경망의 구성에 도움을 줄 수가 있다. 물론 처음에 신경망을 잘못 구성하여 원하는 결과가 나오지 않을 수도 있으나, 이 경우에도 문제를 발생시키는 부분을 찾아 내기가 쉬우므로 재 구성이 용이하다. 또 위의 경우에서 그 패턴의 위치가 조금 바뀐다 하더라도 신경망 전체를 기존의 학습 알고리즘을 이용하여 학습시키면 쉽게 원하는 결과를 얻을 수 있다. 즉 기존의 패턴에서의 작은 변화에 대해서는 쉽게 학습을 할 수 있게 될 것이며, 오차에 대해서도 신경망 전체에 대한 학습데이타의 관찰로 부터 원인을 알아내는 것이 가능할 것이다.

4. 은닉층에 대한 의미 부여를 통한 계산에 적합한 신경망의 구성

일반적인 신경망으로는 더하기 계산이 곤란하다. 그 이유로는 각 노드에서 사용되는 전달 함수(activation function)가 사칙 연산에는 적합하지 않으며, 복잡한 계산을 수행하기 위해 각 노드의 전달함수를 서로 다르게 한다면 신경회로망을 사용하는 의미가 없어질 수도 있을 것이다. 그래서 여기서 제안하고자 하는 전달 함수는 그림 4.1과 같다.

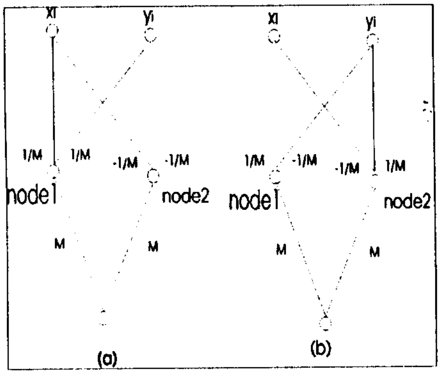


<그림 4.1 사칙연산에 사용되는 전달함수>

$$f(x)=\begin{cases}0 & x<b\\ ax & b<x<b+\frac{1}{a}\\ 1 & x>b+\frac{1}{a}\end{cases} \qquad \text{<식 4.1>}$$

이 함수에서 학습의 대상이 되는 것은 a 과 b 이다. 이러한 전달 함수를 가진 신경망의 덧셈은 아주 간단한 것으로써 그림 4.2와 같다.

$o_i = x_i + y_i$ i 는 정수 $o_i = x_i - y_i$ i 는 정수



<그림 4.2 더하기와 빼기를 수행하는 신경망의 구조>

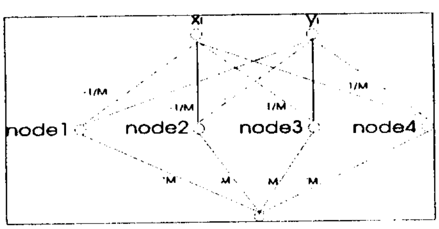
위의 그림4.2(a)에서 노드1의 의미는 $x_i + y_i > 0$ 이고 노드2의 의미는 $x_i + y_i < 0$이다. 그리고 M은 다음의 성질을 만족하여야 한다.

$$M = \max(|x_i|, |y_i|) \quad \text{<식 4.2>}$$

뺄셈 역시 위에서 보여준 것과 같이 더하기의 경우와 비슷한 형태를 이루게 된다.

이제까지 보여준 더하기나 빼기에 비해서 곱셈과 나눗셈은 그 구조에 따라 제한이 발생하게 된다. 여기서 제안하고자 하는 구조는 위의 경우와 마찬가지로 사용되는 숫자가 실수이기만 하면 되는 것이다. 다음으로 곱셈을 수행하는 신경망의 구조이다.

$o_i = x_i * y_i$ i는 정수



<그림 4.3 곱하기를 수행하는 신경망의 구조>

여기에서 볼 수 있듯이 은닉층 즉 두 번째 계층의 전달함수의 a, b, 그리고 M는 다음과 같다.

$$a = |x_i| \text{ or } a = |y_i|, \quad b = 0, \quad M = \max(|x_i|, |y_i|) \quad \text{<식 4.3>}$$

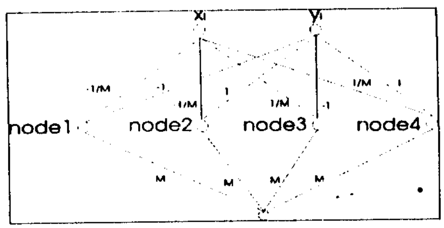
그리고 은닉층의 각 노드의 의미는 다음과 같다.

$$\begin{aligned} node1 &: x_i < 0 \text{ and } y_i < 0 \\ node2 &: x_i < 0 \text{ and } y_i > 0 \\ node3 &: x_i > 0 \text{ and } y_i < 0 \\ node4 &: x_i > 0 \text{ and } y_i > 0 \end{aligned} \quad \text{<식 4.4>}$$

여기서 보인 곱셈을 위한 신경망은 가장 단순한 형태이다. 즉, 이상의 곱셈에 사용되는 신경망은 곱하는 숫자중 하나가 전달 함수내에 반영되어야 하므로 곱하는 숫자에 따라 서로 다른 은닉층을 더 필요로 하게된다. 또 학습 하는 과정에서도 곱하는 숫자 별로 따로 학습이 필요하게 된다. 이러한 신경망은 상당히 복잡한 구조를 가지므로 여기서는 보이지 않는다.

마지막으로 나누기를 수행하는 신경망이다. 조건이나 제반 상황은 곱셈의 경우와 일치한다. 그 구조는 다음과 같다.

$o_i = x_i / y_i$



<그림 4.4 나눗셈을 수행하는 신경망의 구조>

나누기를 하는데 사용되는 신경망의 구조나 전달함수의 성질은 곱셈의 경우와 상당히 유사하다. 그러나 곱셈에서는 곱셈이 사용되어지는 반면에 나눗셈의 경우에는 나누기가 실제로 사용되는 곳이 없게 된다. 여기서 사용되는 은닉층 즉 두 번째 계층의 전달함수의 a, b, 그리고 M는 다음과 같다.

$$a = 0, \quad b = -|y_i / M|, \quad M = \max(|y_i|) \quad \text{<식 4.5>}$$

· 그리고 은닉층의 각 노드의 의미는 다음과 같다.

$$\begin{array}{l} node1 : x_i<0 \text{ and } y_i<0 \\ node2 : x_i<0 \text{ and } y_i>0 \\ node3 : x_i>0 \text{ and } y_i<0 \\ node4 : x_i>0 \text{ and } y_i>0 \end{array} \quad \langle 식\ 4.6\rangle$$

나누기의 경우에도 곱하기와 비슷하게 나누는 숫자에 따라 은닉층의 노드를 설정한다면, 위의 경우에서 발생하는, 나누는 숫자에 따라 전달함수가 달라지는 경우를 배제 할 수 있을 것이다. 그러나 이와 같이 신경망을 구성하게 되면 그 구조는 지나치게 복잡하게 되어 신경망을 사용하는 의미가 없어질 것이다.

이상과 같은 계산에 적합한 신경망의 사용으로 제어등에서 계산을 필요로 하는 부분을 좀더 정확한 출력을 가지도록 할 수 있다. 실제로 나누기를 하는 신경망은 제어에 사용되는 신경망에서 성공적으로 사용되어 졌다.

또 산술연산이 가능하여 짐으로써 신경망에 적합한 하드웨어를 설계했을때 발생할 수 있는 계산의 어려움이 해결될 수도 있을 것이다.

여기서 제안한 사칙연산을 위한 신경망은 실제 신경망의 일부로써 사용되어질 수도 있다. 그리고 신경망으로 구현하기 어려웠던 것을 은닉층에 대한 의미 부여를 통해 실현할 수 있다는 가능성을 제시한다.

결론

지금까지 여러가지 예를 통하여 신경망의 은닉층에 대한 의미부여 방법을 살펴 보았다. 이 방법의 사용으로부터 좀더 빠르고 쉬운 학습을 기대할 수 있을 것이다. 물론 은닉층에 대한 의미부여가 곤란하거나 어려운 경우에는 기존의 신경망을 사용하는 것이 보다 쉬울 것이다. 그러나 신경망 내부의 상세하고 부분적인 작동을 알아볼 수 있다는 점에서는 은닉층에 대한 의미부여가 도움을 줄 수 있을 것이다. 또 기대할 수 있는 것은 이러한 성질이 시스템 해석에서도 사용이 가능해질 수 있다는 것이다.

또 하나 은닉층에 대한 의미부여로써 얻을 수 있을 것이라고 생각되어지는 것은 한 번 학습이된 신경망이 다른 보다 복잡한 신경망에 은닉층으로 포함이 될 수 있다는 것이다. 즉 모듈화가 이루어질 수 있을 것이다. 이러한 성질은 학습시간의 단축에 크게 기여할 수 있을 것이다. 이것은 스파이어럴 문제를 위한 신경망에서 그 예를 보이고 있다.

참고 문헌

1. TAL GROSSNAN, RONNY MEIR & EYTAN DOMANY, "LEARNING BY CHOICE OF INTER REPRESENTATIONS", CONNECTIONIST MODELS SUMMER SCHOOL PROCEEDING OF 1988.

2. 김명찬, "신경회로망을 이용한 문서인식 시스템의 구현", 석사학위논문 서울대학교 1992.

3. RICHARD P. LIPPMANN, "PATTERN CLASSIFICATION USING NEURAL NETWORK", IEEE COMMUNICATION MAGAZINE PP.47-64 1989.

4. GEORGE F. LUGER & WILLIAM A. STUBBLEFIELD, "ARITIFICIAL INTELLIGENT STRUCTURE AND STRATEGIES COMPLEX PROBLEM SOLVING", THE BENJAMIN/CUMMINGS PUBLISHING COMPANY INC., 1992.

5. STEVEN L.TANIMOTO, "THE ELEMENTS OF ARTIFICIAL INTELLIGENCE" COMPUTER SCIENCE PRESS, 1987.

6. S.Y.KUNG, J.N.HWANG, "AN ALGEBRAIC PROJECTION ANALYSIS FOR OPTIMAL HIDDEN UNITS SIZE AND LEARNING RATES IN BACK-PROPAGATION", IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORK, 1988.

7. PAUL R COHEN & EDWARD A. FEIGENBAUM, "THE HANDBOOK OF ARTIFICIAL INTELLIGENCE", ADDISON WESLEY, 1982.

8. KEE SAEHUN, "A COMPARISION OF THE INTELLIGENT SYSTEMS-ARTIFICIAL INTELLIGNECE, NEURAL NETWORK, AND FUZZY", PROCEEDINGS OF ISL WINTERWORKSHOP, VOL.6 1993.

9. LES ATLAS, RONALD COLE, "A PERFORMANCE COMPARISION OF TRAINED MULTILAYER PERCEPTRONS AND TRAINED CLASSIFICATION TREES" PROCEEDING OF THE IEEE, VOL.78, OCTOBER 1990.

10. GAVIN J. GIBSON, COLIN F. N. COWAN, "ON THE DECISION RESIONS OF MULTILAYER PERCEPTRON" PROCEEDING OF THE IEEE, VOL.78, OCTOBER 1990.

11. ALEXIS WIELAND & RUSSELL LIGHTON, "GEOMETRIC ANALYSIS OF NEURAL NETWORK CAPABILITIES",

12. H. J. ZIMMERMANN, L. A. ZADEH & B. G. GAINES, "FUZZY SETS AND DECISION ANALYSIS", NORTH-HOLLAND, 1984.

13. KEVIN J. LANG, MICHAEL J. WITBROCK, "LEARNING TO TELL TWO SPIRALS APART", CONNECTIONIST MODELS SUMMER SCHOOL PROCEEDING OF 1988.